С. Ю. Крюков

ИСПОЛЬЗОВАНИЕ ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛЕНИЙ В ЗАДАЧАХ МАШИННОГО ОБУЧЕНИЯ

Машинное обучение — мощный инструмент обработки больших объемов данных, требующий, однако, выбора между качеством полученных моделей и временем их расчета. Использование параллельных вычислений позволяет ускорить многие алгоритмы.

Введение

Настройка модели алгоритмов по данным — это задача оптимизации, от эффективности решения которой зависит практическая применимость метода машинного обучения. В эпоху больших данных многие классические алгоритмы оптимизации становятся неприменимы, т.к. здесь требуется решать задачи оптимизации функций за время меньшее, чем необходимо для вычисления значения функции в одной точке. Таким требованиям можно удовлетворить в случае грамотного комбинирования известных подходов в оптимизации и использования параллельных вычислений.

I. Математическая модель

Одним из самых простых и, вместе с тем, довольно гибким алгоритмом восстановления зависимости выходных величин по входным является линейная регрессия. В задаче линейной регрессии предполагаемая зависимость между входными значениями и выходной величиной задаётся гипотезой:

$$h(x) = \Theta_0 + \Theta_1 x_1 + \Theta_2 x_2 + ... + \Theta_n x_n = \sum_{n}^{i=0} \Theta_i x_i$$

где Θ_i - неизвестные параметры линейной регрессии (также называемые весами), а для простоты принимается равным 1.

Имея заранее известный набор входных величин и значений выходных параметров от этих величин производится тренировка алгоритма для установления значений Θ_i. Задача сводится к минимизации некой оценочной функции, например, по методу наименьших квадратов:

$$C(\Theta) = \frac{1}{2} \sum_{m}^{i=1} \left(h(x) - y\right)^2$$

Одним из способов эффективного распараллеливания задачи минимизации оценочной функции является её векторизация. Векторизация позволяет заменить итеративный процесс суммирования значений к перемножению двумерных матриц. Существуют эффективные алгоритмы и их практические программные реализации, библиотеки для подобных вычислений, например, алгоритм Кэннона или алгоритм SUMMA (Scalable Universal Matrix Multiplication Algorithm). Но и без применения сложных узкоспециализированных алгоритмов задача линейной регрессии поддаётся распараллеливанию. В векторном виде уравнение относительно Θ может быть переписано в виде:

$$\Theta = \left(X^T X\right)^{-1} X^T y = A_{-1} b$$

где $A = X^T X, b = X^T y$

$$A = \sum_{m}^{i=1} (x_i {x_i}^T),\ b = \sum_{m}^{i=1} (x_i y_i)$$

II. Вывод

Установлено, что A и b могут быть вычислены разными процессами на разных узлах и затем результат может быть обьединён с помощью Map-Reduce фреймворка, такого как Apache Hadoop или Apache Spark.

1. 1. ChengTao Chu. Map-Reduce for Machine Learning on Multicore / Chu, ChengTao, Kyun Kim, Sang // CS. Department, Stanford University, 2013. – 45 с.
2. 2. Lynn Elliot Cannon. A cellular computer to implement the Kalman Filter Algorithm / Cannon, Lynn Elliot // Montana State University, 1969. – 150 с.

Крюков Сергей Юрьевич, аспирант кафедры ИТАС.

Научный руководитель: Навроцкий Анатолий Александрович, заведующий кафедрой автоматизированных систем обработки информации, кандидат физико-математических наук, доцент, navrotsky@bsuir.by.